DRAFT

2.25   DETAILED RESULTS FOR SIGNAL INTEGRATION

The Signal Integration Functional Element does <u>not</u> implement the specified design in Section 2.25 of ASP II. The code quality for the FE is good; however, most algorithms implemented in code to calculate the number of pulses integrated (NPI) had no independent source of corroboration. Internal and external documentation is non-existent.

The table listed below summarizes the desk-checking and software testing verification activities for each subroutine in the Signal Integration Functional Element. The two results columns contain checks if no discrepancies were found. Where discrepancies were found, the desk check results column contains references to discrepancies listed in Table 2.25-4, while the test case results column lists the number of the relevant test case in Table 2.25-6. More detailed information on the results is recorded in those tables.

TABLE 2.25-1. Verification Results Summary for Signal Integration.

Design Element	Code Location	Desk Check Result	Test Case ID	Test Case Result
25-1 Number of Pulses in a Target Return	PDET 60-69	D1 D2	all	all
25-2 Number of Pulses Integrated	PDET 75-92	D3	all	all
User Input	INPUT1, INPUT2	Y	all	Y
System Parameters	RDRDAT	D4	all	25-2

2.25.1   Overview

In many radar systems, returns from multiple pulses are used to make a detection decision. Signal integration is the summation of several successive signal and noise pulses for the purpose of improving the detectability of target signals. Integration gain is the resulting improvement factor in the signal-to-noise ratio.

Integration may be accomplished in the radar receiver either before or after the second (amplifier) detector. Integration before the second detector is called pre-detection or coherent integration, while integration after the second detector is called post-detection or noncoherent integration. Some pre-detection integration methods are addressed in other FEs (clutter rejection in Sections 2.23 and 2.24, and pulse compression in Section 2.26). The FE described in this section includes only post-detection processing. The intent of the Signal Integration FE in *RADGUNS* v.1.8 is to simulate the noncoherent integration performed by human operators for target detection.

The *RADGUNS* implementation of signal integration is done primarily by Subroutine PDET. Initialization routines also are necessary for FE operations. Table 2.25-2 briefly describes the routines. PDET is not dedicated to integration; it calls one other subroutine to implement the Signature Fluctuations FE and the Threshold FE. Only the portion applicable to integration was verified.

DRAFT

TABLE 2.25-2. Subroutine Descriptions.

Module Name	Description
PDET	Calculates number of pulses returned from target and the number of pulses integrated
INPUT1, INPUT2	Initializes data entered in the user-defined parameters file
RDRDAT	Initializes system-specific variables

2.25.2 Design Elements

Design elements defined for the Signal Integration FE are listed in Table 2.25-3. A design element is an algorithm that represents a specific component of the FE design. The first two design elements are fully described in Section 2.25.2 of the ASP II for *RADGUNS*.

TABLE 2.25-3. Integration Design Elements.

Module	Design Element	Description
PDET	25-1: Number of Pulses in a Target Return	Calculates the number of pulses returned from the target for a single radar scan
PDET	25-2: Number of Pulses Integrated	Calculates the number of pulses used in determining the probability of detection of the target
INPUT1, INPUT2	Input	Initializes user-defined input data
RDRDAT	Input	Initializes user-defined input data

2.25.3 Desk Check Activities and Results

The code implementing this FE was manually examined using the procedures described in Section 1.1 of this report. The overall code quality is good, but internal documentation is poor. Discrepancies were discovered during desk checking. Details are described in the following two tables.

TABLE 2.25-4. Code Discrepancies.

Design Element	Desk Check Result
25-1: Number of pulses in a target return	D1. Code observation revealed that calculation of the number of pulses returned from the target (variable NPULSE) used hard-coded numbers rather than variables traceable to a physical source (such as values of radar parameters). The constant values did not match independent calculations based on the applicable variables of ASP II Equations [2.25-1] and [2.25-2].
	D2. For the applicable system, internal documentation described sector and circular search patterns as simulated identically, except that sector search is confined in azimuth. The equations for calculating the number of pulses returned from the target are different for the two search patterns; no independent reference was found to verify the two equations.

TABLE 2.25-4. Code Discrepancies.

Design Element	Desk Check Result
25-2: Number of pulses integrated (NPI)	D3. The only condition that the NPI is set equal to the number of pulses in a target return (if greater than zero) is when the measured target range is <u>greater</u> than the range to the horizon. When the measured target range is less than the range to the horizon, the NPI is always set equal to one for land terrain, and NPI is calculated using ASP II Equation [2.25-3] for sea terrain (for a non-zero number of returned pulses). No independent reference was found to verify the NPI algorithm.
Input, radar system parameters	D4. Subroutine RDRDAT defines radar parameters. However, the value of the pulse repetition interval for the subject system (variable PRIA) is not initialized. A value of zero for the global variable PRIA is passed to Subroutine PDET.

TABLE 2.25-5. Code Quality and Internal Documentation Results.

Subroutine	Code Quality	Internal Documentation
PDET	OK	Internal documentation for the FE is almost non-existent. No formatted header is in PDET, and no descriptive comments are interspersed throughout the module to describe the simulated processes.
INPUT1, INPUT2	OK	Header lacks definition of some input and output variables.
RDRDAT	OK	Header lacks definition of some input and output variables.

2.25.4   Software Test Cases and Results

Software testing for PDET utilized the entire *RADGUNS* model in debug mode. For these tests, *RADGUNS* was run using the subject AAA system, A10A as the target, and EX1.PAR as the parameters file. The weapon system and the target files were used as delivered. Changes made to the parameters file are listed in the test descriptions. The DCL file RUNRG.COM was modified to run *RADGUNS* in VAX debug mode.

DRAFT

TABLE 2.25-6. Software Test Cases for Signal Integration.

Test Case ID	Test Case Description
25-1	OBJECTIVE: Observe variable initialization, and test calculation of NPI for a sector search pattern with a boresight elevation angle < 10 deg on land, with a target flight path located at a range less than that to the horizon. PROCEDURE: 1.　Make the following changes in the EX1 parameters file: Detection Model　　　　　　'PDET' Threshold　　　　　　　　'0.5' Initial Target Position　　'6000.0 250.0 50.0' 2.　Execute *RADGUNS*. 3.　Set a breakpoint at the start of Subroutine PDET, and observe the values of RDRSYS, SCHPAT, VHORO, TERAIN, RHORIZ, BSEL, BSRG. 4.　Observe execution path to variable NPULSE. 5.　Observe the value of NPULSE. 6.　At line 75 examine the values of variables BSRG and RHORIZ. 7.　Observe execution path to variable NPI. 8.　Observe the value of NPI at the end of the subroutine. VERIFY: 1.　The correct initialization of variables occurs in Step 3. 2.　Execution observed in Step 4 branches to line 62. 3.　The value of NPULSE from Step 5 matches independent calculations using ASP II Equation [2.25-1]. 4.　BSRG < RHORIZ using values observed in Step 6. 5.　Execution observed in Step 7 branches to line 84. 6.　The value of NPI from Step 8 matches value of NPULSE observed in Step 5. RESULT: Calculation of the number of pulses returned from the target (variable NPULSE) occurred at the anticipated line number. However, independent calculation using ASP II Equation [2.25-1] did not match the value of NPULSE. The code set the value of NPI to 1, and not to the value of NPULSE.

DRAFT

TABLE 2.25-6. Software Test Cases for Signal Integration. (Contd.)

Test Case ID	Test Case Description
25-2	OBJECTIVE: Observe variable initialization, and test calculation of NPI for both circular search and perfect cueing search patterns at sea, with a boresight elevation angle < 10 degrees, and a target flight path located at a range less than that to the horizon.

PROCEDURE:

1. Make the following changes in the EX1 parameters file:

Search Antenna Pattern	'PERC'
Minimum Search Time	'15.0'
Environment	'SEA'
Sea State	'9'
Wind Aspect	'50'

2. Run *RADGUNS*.

3. Set a breakpoint at the start of Subroutine PDET, and observe the values of PRIA, ISEAST, TERAIN, WLNTHA, BSEL, BSRG, and RHORIZ.

4. Observe the execution path from line 60 to Variable NPULSE.

5. Examine the value of NPULSE.

6. At line 75 examine the values of Variables BSRG and RHORIZ.

7. Examine the values of SIGMAV and ISEAST at line 77.

8. Observe execution path to Variable NPI.

9. Observe the value of NPI at the end of the subroutine.

10. Repeat Steps 1 through 9, except use a circular search pattern with an antenna scan rate of 5 deg/s.

VERIFY:

1. The correct initialization of variables occurs in Step 3.

2. The value of BSEL < 10 degrees.

3. The execution path observed in Step 4 branches to line 67.

4. Value of NPULSE observed in Step 5 matches independent calculations using ASP II Equation [2.25-1].

5. BSRG < RHORIZ using values observed in Step 6.

6. SIGMAV = 0.31xISEAST using values observed in Step 7.

7. The execution path observed in Step 8 branches to line 79.

8. Value of NPI observed in Steps 9 and 10 matches independent calculations using ASP II Equation [2.25-3].

TABLE 2.25-6. Software Test Cases for Signal Integration. (Contd.)

Test Case ID	Test Case Description

25-3	OBJECTIVE: Test NPI calculations for a circular search pattern with a boresight elevation angle > 10 degrees and a target range within the horizon.
	
	PROCEDURE:
	
	1. Make the following changes in the EX1 parameters file:
	
	    Search Antenna Pattern 'CIRC'
	    Azimuth Scan Rate '10'
	
	2. Run *RADGUNS*.
	
	3. Break in subroutine PDET.
	
	4. Examine the values of BSEL, VHORO, BSRG, and RHORIZ at the start of the subroutine.
	
	5. Observe the execution path from line 60 to Variable NPULSE.
	
	6. Observe the value of NPULSE.
	
	7. Observe the execution path to Variable NPI.
	
	8. Observe the value of NPI at the end of the subroutine.
	
	VERIFY:
	
	1. The value of VHORO is 10 deg/s.
	
	2. The value of BSEL > 10 deg.
	
	3. BSRG < RHORIZ using values observed in Step 4.
	
	4. The execution path observed in Step 5 branches to line 65.
	
	5. The value of NPULSE observed in Step 6 matches independent calculations using ASP II Equation [2.25-2].
	
	6. The execution path observed in Step 7 branches to line 84.
	
	7. The value of NPI observed in Step 8 matches the value of NPULSE observed in Step 6.
	
	RESULT: Calculation of the number of pulses returned from the target (variable NPULSE) occurred at the anticipated line number. However, independent calculation using ASP II Equation [2.25-2] did not match the value of NPULSE. The code set the value of NPI to 1, and not to the value of NPULSE.

TABLE 2.25-6. Software Test Cases for Signal Integration. (Contd.)

Test Case ID	Test Case Description
25-4	OBJECTIVE: Test NPI calculations for a circular search pattern with a boresight elevation angle > 10 degrees and a target range greater than that to the horizon. PROCEDURE: 1. Repeat Step 1 from Test 25-3 with the following changes: Initial Target Position '10000 250 50' 2. Run *RADGUNS*. 3. Set a breakpoint at the start of Subroutine PDET, and examine the values of BSRG and RHORIZ. 4. Observe execution path to variable NPULSE. 5. Observe the value of NPULSE. 6. Observe execution path to variable NPI. 7. Observe the value of NPI at the end of the subroutine. 8. Repeat Steps 1 through 7, except use an azimuth scan rate of 35. VERIFY: 1. BSRG > RHORIZ using values observed in Step 3. 2. The execution path observed in Step 4 branches to line 65. 3. Value of NPULSE observed in Step 5 matches independent calculations using ASP II Equation [2.25-2]. 4. The execution path observed in Steps 6 and 8 branches to lines 88 and 90, respectively. 5. Value of NPI observed in Steps 7 and 8 matches the value of NPULSE observed in Step 5. RESULT: Calculation of the number of pulses returned from the target (variable NPULSE) occurred at the anticipated line number. However, independent calculation using ASP II Equation [2.25-2] did not match the value of NPULSE. The second simulation run resulted in one pulse integrated for zero pulses returned from the target.

2.25.5   Conclusions and Recommendations

2.25.5.1   Code Discrepancies

The product of $q_B \cdot f_p$ in the ASP II Equations [2.25-1] and [2.25-2] is implemented as a constant in the *RADGUNS* code. The constant does not match the product $q_B \cdot f_p$ as defined in the input subroutine (RDRDAT). Furthermore, the implementation of Equation [2.25-1] and [2.25-2] for sector and circular search patterns are different, and neither of the constants in the two equations can be duplicated using the radar data in RDRDAT. These discrepancies should be resolved.

The number of pulses returned from the target (calculated in Blocks 2, 4, and 5 of ASP II Figure 2.25-2) generally should be greater than one. Only one pulse is integrated for all land forms, however, if the apparent range of the target is less than the range of the horizon. The only condition in *RADGUNS* for the NPI to be set to the number of pulses returned from the target is when the target range is greater than the range to the horizon. The code for determining the number of pulses returned from the target and the logic used to set the value of NPI is recommended to be replaced with verifiable equations using variables which define system parameters. Also, the correct PRI for acquisition mode should be initialized in Subroutine RDRDAT, Variable PRIA.

2.25.5.2   Code Quality and Internal Documentation

Code quality is generally good for Subroutine PDET, which implements the FE. Internal documentation is deficient. A formatted header should be added to the subroutine, and descriptive comments should be added throughout the module to describe the simulated processes. Variable declaration is complete for subroutine arguments and global variables, but local variable declarations should be added to PDET. Initialization of local variables also should be added to the module.

2.25.5.3   External Documentation

External documentation for the FE is non-existent; description is entirely omitted from the combined User/Analyst/Programmer Manual for *RADGUNS*. The combined manual has analyst's information included in appendices that present theoretical descriptions of several modeled processes; however, the methodology for signal integration modeling is not presented in the manual. Integration is intrinsically interwoven with the Signature Fluctuations FE (Section 2.4) and the Threshold FE (Section 2.22); these two FEs and the Signal Integration FE comprise the probability of detection model, which should be explained in a separate appendix; Appendix VII is unused, and should be utilized to enhance the document completeness.

The programmer's manual information is documented in the form of module descriptions. File RGPDET.FOR contains five subroutines which implement the probability of detection model (only Subroutine PDET applies directly to the Integration FE). These modules are omitted entirely from description in the manual. Thus, their descriptions should be included in Section V, Subroutine and Function Descriptions.